ARMY RESEARCH LABORATORY

Genetic Programming-based Phononic Bandgap Structure Design

by Raymond A. Wildman and George A. Gazonas

ARL-TR-5733 **September 2011**

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory

Aberdeen Proving Ground, MD 21005-5066

ARL-TR-5733 **September 2011**

Genetic Programming-based Phononic Bandgap Structure Design

Raymond A. Wildman and George A. Gazonas

Weapons and Materials Research Directorate, ARL

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
September 2011	Final	October 2009-July 2011

4. TITLE AND SUBTITLE
Genetic Programming-based Phononic Bandgap Structure Design

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Raymond A. Wildman
George A. Gazonas

5d. PROJECT NUMBER
AH80

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
ATTN: RDRL-WMM-B
Aberdeen Proving Ground, MD 21005-5069

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-5733

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
primary author's email: <raymond.a.wildman.civ@mail.mil>

14. ABSTRACT
Two-dimensional phononic bandgap materials are designed using a genetic programming topology optimization method and a finite element elastic wave solver. The optimization problem involves maximizing the bandgap, or range of blocked frequencies of propagating elastic waves, in a periodic structure by designing the shape of an inclusion. This problem is modeled as a single unit cell using the time-harmonic elastodynamic wave equation with Floquet (periodic) boundary conditions. After discretization, an eigenvalue solver is used to compute the allowed frequencies of propagation for a certain wave vector. The geometry optimization method uses a tree structure to define geometry: internal tree nodes represent a priority-based overlap and leaf nodes contain a list of points whose convex hull represent a convex polygon. A genetic programming method is used to optimize this data structure. Several bandgap structures are designed using different materials, unit cell shapes, and total number of available materials. The results show that bandgaps exist for several different material systems though typically not just between the first and second bands. In addition to the inclusion shape, the size of the bandgap usually depends on the materials, with materials systems having large differences in wave speeds producing larger gaps.

15. SUBJECT TERMS
phononic bandgap; genetic programming; optimization; geometry

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Raymond A. Wildman
Unclassified	Unclassified	Unclassified	UU	36	19b. TELEPHONE NUMBER (*Include area code*) 410-306-2232

Contents

List of Figures

List of Tables

1. Introduction

Phononic bandgap materials—periodic materials that block acoustic or elastic wave propagation in a range of frequencies—have a variety of applications including piezoelectric transducers (*1*), ultrasound and polarization filters, acoustic filters for sound reduction, and vibration-less environments for sensitive equipment (*2*). There has been recent research into designing such materials, first dating back to 1994 (*3*), which showed the design of bandgap materials by varying the filling fraction of a specified geometry. Later, the actual geometries of bandgap material inclusions were optimized (*4*). Finally, in reference 5, genetic algorithms were used to maximize the bandgap of phononic materials by designing an inclusion.

In previous work, all methods used a discretized unit cell, either with square elements, as in reference 4, or triangular elements, as in reference 5. A major downside of using a strict grid is that the grid resolution can be critical in outcome and efficiency. Using a too coarse grid limits the complexity of the solutions, but a too fine grid may take exceedingly long to converge, if at all. Another issue in references 4 and 5 was that both approaches only considered two-phase systems. While not a severe limitation, it may be fruitful to allow the optimizer to choose which materials to use in a design, regardless of number. Finally, reference 5 only considered acoustic wave propagation, not full elastic waves.

These limitations are addressed here, specifically by using a genetic programming approach to geometry optimization, introduced in reference 6, and extended in references 7 and 8 and using the elastic wave equation in place of the acoustic equation. The genetic programming approach represents geometry using a tree structure, representing combinations of arbitrary convex polygons. In this tree structure, each leaf node contains a list of points and the convex hull of this list of points represents a convex polygon, to be combined with other polygons as defined in the function nodes of the tree. The lists themselves are not limited in size, so each convex polygon can have any number of vertices. This approach is naturally hierarchical in contrast to strict grid methods, as in references 4 and 5. The method can be initialized with a single convex polygon with a small number of vertices, while more polygons and vertices can be inserted later. For multi-phase systems, material parameters can be inserted in the leaf nodes, as in reference 8. As is shown, lifting these limitations leads to higher quality designs, more efficiently.

The remainder of this report describes the systematic design of phononic bandgap materials using genetic programming. Specifically, section 2 gives the formulation of the problem in terms of

finite elements and discusses a method for determining the bandgap size for a given material. Section 3 describes the geometry optimization method, which uses genetic programming and convex polygons to construct a solution. Several optimization results are presented in section 4 and the conclusions are discussed in section 5.

2. Phononic Bandgap Material Model

The main purpose of this work is to lift many of the limitations in place in reference 5. First, reference 5 only considered scalar dilatational waves; here the full vector, elastic problem is treated. Another restriction was the limit to two-phase systems, which is replaced with a more general method capable of choosing from a database of materials.

Phononic bandgap materials are periodic structures, defined by some unit cell. The unit cell of the problem considered is shown in figure 1 and has a few defining parameters: the side lengths, $\ell_i = |\mathbf{l}_i|$, and the acute angle α. Figure 1 is labeled with the edge vectors $\mathbf{l}_i$, though typically the problem is defined using side lengths and acute angle via

$$\begin{aligned} \mathbf{l}_1 &= \ell_1 \hat{\mathbf{x}}, \\ \mathbf{l}_2 &= \ell_2 \cos(\alpha)\hat{\mathbf{x}} + \ell_2 \sin(\alpha)\hat{\mathbf{y}}. \end{aligned} \tag{1}$$

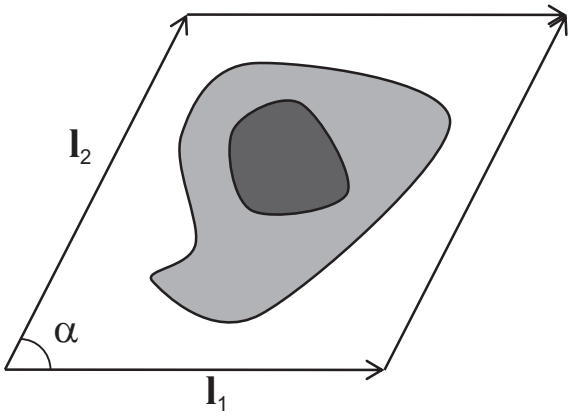

Figure 1. Unit cell diagram.

In a phononic bandgap material, assuming it is infinitely periodic and there are no body forces, the displacement $\mathbf{u} = u^{\mathrm{x}}\hat{\mathbf{x}} + u^{\mathrm{y}}\hat{\mathbf{y}}$ follows the elastic wave equation

$$\begin{aligned}\rho\frac{\partial^2 u^{\mathrm{x}}}{\partial t^2} &= \frac{\partial}{\partial x}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{x}}}{\partial x} + \lambda\frac{\partial u^{\mathrm{y}}}{\partial y}\right] + \frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u^{\mathrm{x}}}{\partial y} + \frac{\partial u^{\mathrm{y}}}{\partial x}\right)\right], \\ \rho\frac{\partial^2 u^{\mathrm{y}}}{\partial t^2} &= \frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u^{\mathrm{y}}}{\partial x} + \frac{\partial u^{\mathrm{x}}}{\partial y}\right)\right] + \frac{\partial}{\partial y}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{y}}}{\partial y} + \lambda\frac{\partial u^{\mathrm{x}}}{\partial x}\right],\end{aligned} \tag{2}$$

where λ and μ are the Lamé parameters. Since we are interested in time-harmonic solutions, equation 2 can be converted to the frequency domain assuming $e^{j\omega t}$ time dependence:

$$\begin{aligned}\rho\omega^2 u^{\mathrm{x}} &= -\frac{\partial}{\partial x}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{x}}}{\partial x} + \lambda\frac{\partial u^{\mathrm{y}}}{\partial y}\right] - \frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u^{\mathrm{x}}}{\partial y} + \frac{\partial u^{\mathrm{y}}}{\partial x}\right)\right], \\ \rho\omega^2 u^{\mathrm{y}} &= -\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u^{\mathrm{y}}}{\partial x} + \frac{\partial u^{\mathrm{x}}}{\partial y}\right)\right] - \frac{\partial}{\partial y}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{y}}}{\partial y} + \lambda\frac{\partial u^{\mathrm{x}}}{\partial x}\right].\end{aligned} \tag{3}$$

The appropriate boundary conditions must now be formulated to reduce the problem to a single unit cell. As the problem is infinitely periodic, we assume that the solution will not grow or shrink as it travels through the material. (Were it to grow or shrink, it would either grow to infinity or shrink to zero, neither of which is useful.) Assuming the magnitude does not change across unit cells, the solutions on opposing boundaries must then be phase-shifted versions, also known as Floquet boundary conditions (*9*), which can be written as

$$\begin{aligned}\mathbf{u}(\mathbf{r}+\mathbf{l}_1) &= \mathbf{u}(\mathbf{r})e^{j\mathbf{k}\cdot\mathbf{l}_1},\ \mathbf{r} = a\mathbf{l}_2, \\ \mathbf{u}(\mathbf{r}+\mathbf{l}_2) &= \mathbf{u}(\mathbf{r})e^{j\mathbf{k}\cdot\mathbf{l}_2},\ \mathbf{r} = b\mathbf{l}_1, \\ \nabla\cdot\mathbf{u}(\mathbf{r}+\mathbf{l}_1) &= \nabla\cdot\mathbf{u}(\mathbf{r})e^{j\mathbf{k}\cdot\mathbf{l}_1},\ \mathbf{r} = a\mathbf{l}_2, \\ \nabla\cdot\mathbf{u}(\mathbf{r}+\mathbf{l}_2) &= \nabla\cdot\mathbf{u}(\mathbf{r})e^{j\mathbf{k}\cdot\mathbf{l}_2},\ \mathbf{r} = b\mathbf{l}_1,\end{aligned} \tag{4}$$

where $0 \leq a, b \leq 1$ are arbitrary real numbers.

Equations 3 and 4 define a generalized eigenvalue problem in ω and $\mathbf{u}$, given a wave-vector $\mathbf{k}$. Because of the periodicity of the problem, solutions will occur in discrete modes, so that for a given $\mathbf{k}$, a discrete set of allowed frequencies of propagation ω can be found. If these equations can be solved for all $\mathbf{k}$, the full set of allowed propagation frequencies can be recovered. Regions of no allowed propagation can exist, which defines a bandgap. Fortunately, the solutions themselves are periodic in $\mathbf{k}$, so only a finite region must be searched for bandgaps. The defining

region of **k**, or Brillouin zone, is related to the edge vectors **l** by

$$\mathbf{k}_m \cdot \mathbf{l}_n = 2\pi\delta_{mn}, \tag{5}$$

where δ_{mn} is the Kronecker delta function. This set of equations can now be discretized using finite elements and solved as an eigenvalue problem to find the size of the bandgap for a given phononic bandgap material. This process is described in the following subsections.

2.1 Finite Element Formulation

First, the governing equations are converted to a weak form, by projecting the equations onto a set of testing functions

$$\begin{aligned}
\omega^2 \int_\Omega \rho t_m^{\mathrm{x}} u^{\mathrm{x}} \mathbf{dr} &= -\int_\Omega t_m^{\mathrm{x}} \frac{\partial}{\partial x}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{x}}}{\partial x} + \lambda\frac{\partial u^{\mathrm{y}}}{\partial y}\right]\mathbf{dr} - \int_\Omega t_m^{\mathrm{x}} \frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u^{\mathrm{x}}}{\partial y} + \frac{\partial u^{\mathrm{y}}}{\partial x}\right)\right]\mathbf{dr}, \\
0 &= -\int_\Omega t_m^{\mathrm{x}} \frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u^{\mathrm{y}}}{\partial x} + \frac{\partial u^{\mathrm{x}}}{\partial y}\right)\right]\mathbf{dr} - \int_\Omega t_m^{\mathrm{x}} \frac{\partial}{\partial y}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{y}}}{\partial y} + \lambda\frac{\partial u^{\mathrm{x}}}{\partial x}\right]\mathbf{dr}, \\
0 &= -\int_\Omega t_m^{\mathrm{y}} \frac{\partial}{\partial x}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{x}}}{\partial x} + \lambda\frac{\partial u^{\mathrm{y}}}{\partial y}\right]\mathbf{dr} - \int_\Omega t_m^{\mathrm{y}} \frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u^{\mathrm{x}}}{\partial y} + \frac{\partial u^{\mathrm{y}}}{\partial x}\right)\right]\mathbf{dr}, \\
\omega^2 \int_\Omega \rho t_m^{\mathrm{y}} u^{\mathrm{y}} \mathbf{dr} &= -\int_\Omega t_m^{\mathrm{y}} \frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u^{\mathrm{y}}}{\partial x} + \frac{\partial u^{\mathrm{x}}}{\partial y}\right)\right]\mathbf{dr} - \int_\Omega t_m^{\mathrm{y}} \frac{\partial}{\partial y}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{y}}}{\partial y} + \lambda\frac{\partial u^{\mathrm{x}}}{\partial x}\right]\mathbf{dr},
\end{aligned} \tag{6}$$

where t_m^{x} and t_m^{y} are the x- and y-components of a vector testing function $\mathbf{t}_m$ and Ω denotes the unit cell. The double derivatives on the displacement in equation 6 can be removed by shifting the derivatives onto the testing functions. After some simplification, we get

$$\begin{aligned}
\omega^2 \int_\Omega \rho t_m^{\mathrm{x}} u^{\mathrm{x}} \mathbf{dr} &= \int_\Omega \frac{\partial t_m^{\mathrm{x}}}{\partial x}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{x}}}{\partial x} + \lambda\frac{\partial u^{\mathrm{y}}}{\partial y}\right]\mathbf{dr} + \int_\Omega \frac{\partial t_m^{\mathrm{x}}}{\partial y}\left[\mu\left(\frac{\partial u^{\mathrm{x}}}{\partial y} + \frac{\partial u^{\mathrm{y}}}{\partial x}\right)\right]\mathbf{dr}, \\
0 &= \int_\Omega \frac{\partial t_m^{\mathrm{x}}}{\partial x}\left[\mu\left(\frac{\partial u^{\mathrm{y}}}{\partial x} + \frac{\partial u^{\mathrm{x}}}{\partial y}\right)\right]\mathbf{dr} + \int_\Omega \frac{\partial t_m^{\mathrm{x}}}{\partial y}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{y}}}{\partial y} + \lambda\frac{\partial u^{\mathrm{x}}}{\partial x}\right]\mathbf{dr}, \\
0 &= \int_\Omega \frac{\partial t_m^{\mathrm{y}}}{\partial x}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{x}}}{\partial x} + \lambda\frac{\partial u^{\mathrm{y}}}{\partial y}\right]\mathbf{dr} + \int_\Omega \frac{\partial t_m^{\mathrm{y}}}{\partial y}\left[\mu\left(\frac{\partial u^{\mathrm{x}}}{\partial y} + \frac{\partial u^{\mathrm{y}}}{\partial x}\right)\right]\mathbf{dr}, \\
\omega^2 \int_\Omega \rho t_m^{\mathrm{y}} u^{\mathrm{y}} \mathbf{dr} &= \int_\Omega \frac{\partial t_m^{\mathrm{y}}}{\partial x}\left[\mu\left(\frac{\partial u^{\mathrm{y}}}{\partial x} + \frac{\partial u^{\mathrm{x}}}{\partial y}\right)\right]\mathbf{dr} + \int_\Omega \frac{\partial t_m^{\mathrm{y}}}{\partial y}\left[(\lambda+2\mu)\frac{\partial u^{\mathrm{y}}}{\partial y} + \lambda\frac{\partial u^{\mathrm{x}}}{\partial x}\right]\mathbf{dr}.
\end{aligned} \tag{7}$$

Next, the unit cell is meshed with triangular elements using the method of Persson and Strang (*10*). One important note regarding the meshing is that the nodes on opposing boundary edge pairs should be equally spaced and equal in number to facilitate the application of the boundary

conditions. Given this discretization, the solution is approximated with a set of basis functions, which are identical to the testing functions. The basis and testing functions used here are linear, nodal elements, which are unity at a node and zero at all surrounding mesh edges. (This function is also known as a hat function.) Expanding the solution in a series and separating the boundary terms gives

$$\begin{aligned} u^{\mathrm{x}}(\mathbf{r}) &\approx \sum_{n=1}^{N} v_n^{\mathrm{x}} b_n^{\mathrm{x}}(\mathbf{r}) + \sum_{n=1}^{M} w_n^{\mathrm{x}} b_{n+N}^{\mathrm{x}}(\mathbf{r}) + \sum_{n=1}^{P} z_n^{\mathrm{x}} b_{n+N+M}^{\mathrm{x}}(\mathbf{r}), \\ u^{\mathrm{y}}(\mathbf{r}) &\approx \sum_{n=1}^{N} v_n^{\mathrm{y}} b_n^{\mathrm{y}}(\mathbf{r}) + \sum_{n=1}^{M} w_n^{\mathrm{y}} b_{n+N}^{\mathrm{y}}(\mathbf{r}) + \sum_{n=1}^{P} z_n^{\mathrm{y}} b_{n+N+M}^{\mathrm{y}}(\mathbf{r}), \end{aligned} \tag{8}$$

where u^{x} and u^{y} are the x- and y-components of the displacement, b_n^{x} and b_n^{y} are the x- and y-components of a vector basis function $\mathbf{b}_n$, N is the number of internal nodes, M is the number of independent boundary nodes, and P is the number of dependent boundary nodes. The difference between independent and dependent nodes can be gleaned from equation 4: A dependent boundary node is a node that can be written in terms of the solution at another boundary node. We define the independent boundary nodes to be those nodes along $\mathbf{l}_1$ and $\mathbf{l}_2$, except at the $\mathbf{r} = \mathbf{l}_1$ and $\mathbf{r} = \mathbf{l}_2$ corners. Applying the Floquet boundary conditions to the unknown boundary term coefficients gives

$$\begin{aligned} z_p^{\mathrm{x}} &= e^{j\mathbf{k}\cdot\mathbf{l}_1} w_n^{\mathrm{x}},\ \mathbf{r}_p = \mathbf{r}_n + \mathbf{l}_1, \\ z_p^{\mathrm{x}} &= e^{j\mathbf{k}\cdot\mathbf{l}_2} w_n^{\mathrm{x}},\ \mathbf{r}_p = \mathbf{r}_n + \mathbf{l}_2, \\ z_p^{\mathrm{y}} &= e^{j\mathbf{k}\cdot\mathbf{l}_1} w_n^{\mathrm{y}},\ \mathbf{r}_p = \mathbf{r}_n + \mathbf{l}_1, \\ z_p^{\mathrm{y}} &= e^{j\mathbf{k}\cdot\mathbf{l}_2} w_n^{\mathrm{y}},\ \mathbf{r}_p = \mathbf{r}_n + \mathbf{l}_2, \end{aligned} \tag{9}$$

for nodes located on the appropriate boundary according to equation 4. These relations can be collected into matrix form

$$\begin{aligned} \mathbf{z}^{\mathrm{x}} &= \mathbf{D}^{\mathrm{x}} \mathbf{w}^{\mathrm{x}}, \\ \mathbf{z}^{\mathrm{y}} &= \mathbf{D}^{\mathrm{y}} \mathbf{w}^{\mathrm{y}}. \end{aligned} \tag{10}$$

The matrices $\mathbf{D}^{\mathrm{x}}$ and $\mathbf{D}^{\mathrm{y}}$ contain the phase shift terms from equation 9, one coefficient per

column. After substituting the approximate solution from equation 8 into equation 7, we get

$$\omega^2 \begin{bmatrix} \mathbf{B}_{vv}^{xx} & \mathbf{B}_{vw}^{xx} & \mathbf{B}_{vz}^{xx} & & & \\ \mathbf{B}_{wv}^{xx} & \mathbf{B}_{ww}^{xx} & \mathbf{B}_{wz}^{xx} & & \mathbf{0} & \\ \mathbf{B}_{zv}^{xx} & \mathbf{B}_{zw}^{xx} & \mathbf{B}_{zz}^{xx} & & & \\ & & & \mathbf{B}_{vv}^{yy} & \mathbf{B}_{vw}^{yy} & \mathbf{B}_{vz}^{yy} \\ & \mathbf{0} & & \mathbf{B}_{wv}^{yy} & \mathbf{B}_{ww}^{yy} & \mathbf{B}_{wz}^{yy} \\ & & & \mathbf{B}_{zv}^{yy} & \mathbf{B}_{zw}^{yy} & \mathbf{B}_{zz}^{yy} \end{bmatrix} \begin{bmatrix} \mathbf{v}^x \\ \mathbf{w}^x \\ \mathbf{z}^x \\ \mathbf{v}^y \\ \mathbf{w}^y \\ \mathbf{z}^y \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{vv}^{xx} & \mathbf{A}_{vw}^{xx} & \mathbf{A}_{vz}^{xx} & \mathbf{A}_{vv}^{xy} & \mathbf{A}_{vw}^{xy} & \mathbf{A}_{vz}^{xy} \\ \mathbf{A}_{wv}^{xx} & \mathbf{A}_{ww}^{xx} & \mathbf{A}_{wz}^{xx} & \mathbf{A}_{wv}^{xy} & \mathbf{A}_{ww}^{xy} & \mathbf{A}_{wz}^{xy} \\ \mathbf{A}_{zv}^{xx} & \mathbf{A}_{zw}^{xx} & \mathbf{A}_{zz}^{xx} & \mathbf{A}_{zv}^{xy} & \mathbf{A}_{zw}^{xy} & \mathbf{A}_{zz}^{xy} \\ \mathbf{A}_{vv}^{yx} & \mathbf{A}_{vw}^{yx} & \mathbf{A}_{vz}^{yx} & \mathbf{A}_{vv}^{yy} & \mathbf{A}_{vw}^{yy} & \mathbf{A}_{vz}^{yy} \\ \mathbf{A}_{wv}^{yx} & \mathbf{A}_{ww}^{yx} & \mathbf{A}_{wz}^{yx} & \mathbf{A}_{wv}^{yy} & \mathbf{A}_{ww}^{yy} & \mathbf{A}_{wz}^{yy} \\ \mathbf{A}_{zv}^{yx} & \mathbf{A}_{zw}^{yx} & \mathbf{A}_{zz}^{yx} & \mathbf{A}_{zv}^{yy} & \mathbf{A}_{zw}^{yy} & \mathbf{A}_{zz}^{yy} \end{bmatrix} \begin{bmatrix} \mathbf{v}^x \\ \mathbf{w}^x \\ \mathbf{z}^x \\ \mathbf{v}^y \\ \mathbf{w}^y \\ \mathbf{z}^y \end{bmatrix}. \tag{11}$$

Each submatrix in the matrices above correspond to specific terms of equation 7 after basis function substitution. The terms of $\mathbf{B}_{vv}^{xx}$ equate to

$$B_{vv,mn}^{xx} = \int_{\Omega} \rho t_m^x b_n^x \mathbf{dr}, \tag{12}$$

and $\mathbf{A}_{vv}^{xx}$ is given by

$$A_{vv,mn}^{xx} = \int_{\Omega} \left[(\lambda + 2\mu) \frac{\partial t_m^x}{\partial x} \frac{\partial b_n^x}{\partial x} + \mu \frac{\partial t_m^x}{\partial y} \frac{\partial b_n^x}{\partial y} \right] \mathbf{dr}, \tag{13}$$

e.g.

The dependent boundary terms can be eliminated by substituting equation 10 into equation 11,

arriving at

$$\omega^2 \begin{bmatrix} \mathbf{B}_{vv}^{xx} & \mathbf{B}_{vw}^{xx} + \mathbf{B}_{vz}^{xx}\mathbf{D} & \mathbf{0} & \mathbf{0} \\ \mathbf{B}_{wv}^{xx} + \mathbf{B}_{zv}^{xx}\mathbf{D} & \left(\begin{matrix}\mathbf{B}_{ww}^{xx} + \mathbf{B}_{wz}^{xx}\mathbf{D} + \\ \mathbf{D}^*\mathbf{B}_{zw}^{xx} + \mathbf{D}^*\mathbf{B}_{zz}^{xx}\mathbf{D}\end{matrix}\right) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B}_{vv}^{yy} & \mathbf{B}_{vw}^{yy} + \mathbf{B}_{vz}^{yy}\mathbf{D} \\ \mathbf{0} & \mathbf{0} & \mathbf{B}_{wv}^{yy} + \mathbf{B}_{zv}^{yy}\mathbf{D} & \left(\begin{matrix}\mathbf{B}_{ww}^{yy} + \mathbf{B}_{wz}^{yy}\mathbf{D} + \\ \mathbf{D}^*\mathbf{B}_{zw}^{yy} + \mathbf{D}^*\mathbf{B}_{zz}^{yy}\mathbf{D}\end{matrix}\right) \end{bmatrix} \begin{bmatrix} \mathbf{v}^x \\ \mathbf{w}^x \\ \mathbf{v}^y \\ \mathbf{w}^y \end{bmatrix} =$$
$$\begin{bmatrix} \mathbf{A}_{vv}^{xx} & \mathbf{A}_{vw}^{xx} + \mathbf{A}_{vz}^{xx}\mathbf{D} & \mathbf{A}_{vv}^{xy} & \mathbf{A}_{vw}^{xy} + \mathbf{A}_{vz}^{xy}\mathbf{D} \\ \mathbf{A}_{wv}^{xx} + \mathbf{A}_{zv}^{xx}\mathbf{D} & \left(\begin{matrix}\mathbf{A}_{ww}^{xx} + \mathbf{A}_{wz}^{xx}\mathbf{D} + \\ \mathbf{D}^*\mathbf{A}_{zw}^{xx} + \mathbf{D}\mathbf{A}_{zz}^{xx}\mathbf{D}\end{matrix}\right) & \mathbf{A}_{wv}^{xy} + \mathbf{A}_{zv}^{xy}\mathbf{D} & \left(\begin{matrix}\mathbf{A}_{ww}^{xy} + \mathbf{A}_{wz}^{xy}\mathbf{D} + \\ \mathbf{D}^*\mathbf{A}_{zw}^{xy} + \mathbf{D}^*\mathbf{A}_{zz}^{xy}\mathbf{D}\end{matrix}\right) \\ \mathbf{A}_{vv}^{yx} & \mathbf{A}_{vw}^{yx} + \mathbf{A}_{vz}^{yx}\mathbf{D} & \mathbf{A}_{vv}^{yy} & \mathbf{A}_{vw}^{yy} + \mathbf{A}_{vz}^{yy}\mathbf{D} \\ \mathbf{A}_{wv}^{yy} + \mathbf{A}_{zv}^{yy}\mathbf{D} & \left(\begin{matrix}\mathbf{A}_{ww}^{yy} + \mathbf{A}_{wz}^{yy}\mathbf{D} + \\ \mathbf{D}^*\mathbf{B}_{zw}^{yx} + \mathbf{D}^*\mathbf{A}_{zz}^{yx}\mathbf{D}\end{matrix}\right) & \mathbf{A}_{wv}^{yy} + \mathbf{A}_{zv}^{yy}\mathbf{D} & \left(\begin{matrix}\mathbf{A}_{ww}^{yy} + \mathbf{A}_{wz}^{yy}\mathbf{D} + \\ \mathbf{D}^*\mathbf{B}_{zw}^{yy} + \mathbf{D}^*\mathbf{A}_{zz}^{yy}\mathbf{D}\end{matrix}\right) \end{bmatrix} \begin{bmatrix} \mathbf{v}^x \\ \mathbf{w}^x \\ \mathbf{v}^y \\ \mathbf{w}^y \end{bmatrix}. \tag{14}$$

Equation 14 is a generalized eigenvalue problem of the form

$$\omega^2 \mathbf{B}\mathbf{u} = \mathbf{A}\mathbf{u}. \tag{15}$$

As in reference 5, ARPACK is used to solve the eigenvalue problem. Using the fact that **B** is Hermitian, positive definite, **B** can be decomposed using a Cholesky factorization, and equation 15 can be solved more efficiently. Ultimately, for a given **k** vector (which determines the matrix **D**), the eigenvalues ω^2 represent the allowed frequencies of propagation for that given wave vector. Knowledge of ω given **k** is necessary in computing the bandgap of a material as is discussed in the next subsection.

2.2 Bandgap Determination

The finite element formulation is only the first step of determining the bandgap of a structure as any single solution of the above problem only reveals the propagation frequencies and modes at a single **k** vector. A band diagram can be constructed by solving the problem at different values of **k**, and the bandgap can be approximated visually. Automating this process requires a local search method to determine the maximum frequency of a one band and the minimum frequency of the next highest band. In reference 5, the bandgap always appeared between the first and second bands of the structure; however, because they are composed of both longitudinal and shear components, elastic waves generate bandgaps between higher bands, typically the third and

fourth, but never the first and second. Figure 2 shows an example of a band diagram: Note that the bandgap appears between the third and fourth bands. Additionally, the band structure shown in figure 2 has mirror symmetry about the $k_{\mathrm{x}} = k_{\mathrm{y}}$ line.

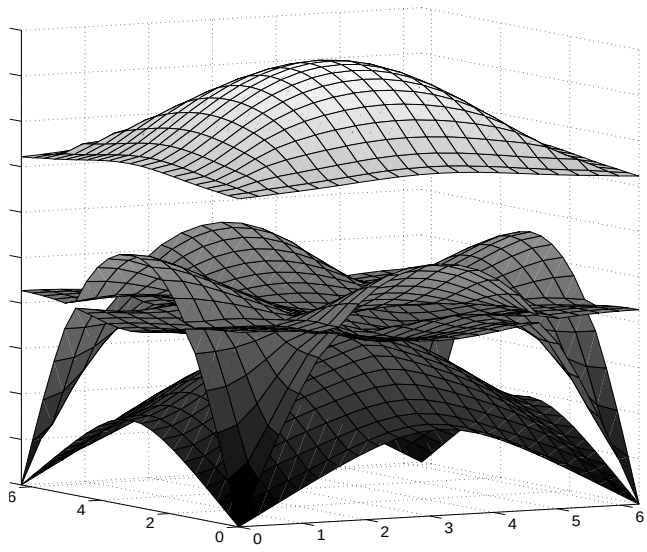

Figure 2. Example of a band diagram, generated from a silicone rubber/lead system with a square inclusion in a square unit cell.

Overall, the approach is similar to that of reference 5, with a few changes. First, the relative bandgap is computed by finding the maximum frequency (versus **k**) in the first band, ω_1, and the minimum frequency in the second band, ω_2. Given these quantities, the relative bandgap is

$$b_{\text{relative}} = \frac{\omega_2 - \omega_1}{\sqrt{\omega_1 \omega_2}} \tag{16}$$

Previously, a simplex method was used to determine the relative bandgap. The simplex method is robust and fairly efficient, though it is not always the most accurate method. (It also may not be more efficient than a derivative-based method with a good starting guess.) Here we use a standard derivative-based method, the Broyden-Fletcher-Goldfarb-Shanno method (BFGS) (*11*). One issue with derivative-based methods is that they require a good starting location to find the global minimum of a function. As can be seen from figure 2, there are many trouble areas for local search methods. Instead of using a single, random starting vector, a uniform 5-by-5 grid of points is mapped in **k**-space and the bandgap is evaluated at only the points that (inclusively) fall

to the left of the $k_x = k_y$ line. The best initial guess of the 15 points is used as the starting point for BFGS, and initial guesses for both bands of interest can be computed simultaneously. Although this approach requires 15 function evaluations initially, it typically leads to fast and robust convergence of BFGS.

---

3. Geometry Optimization Method

Phononic bandgap materials may be composed of simply a matrix with a single inclusion, or possibly multiple materials each with arbitrary topology. The method outlined in reference 5 considered only two-phase systems and used a fixed triangular grid to represent geometry. The main issue with the approach of reference 5 was the fixed discretization, which requires more optimization parameters to return a higher resolution solution. An increase in optimization parameters leads to longer run times, and possibly, poor convergence. Here, a more flexible method based on genetic programming first discussed in reference 6 is used with minor modifications. The main advantage over a fixed discretization is that large, homogeneous regions can be represented with far fewer parameters. The genetic programming approach is also naturally hierarchical because points and polygons can be added to chromosomes as needed. This method has the advantages of a finer discretization, but without the need to overestimate the required grid size. Subsection 3.1 reviews the geometry encoding and subsection 3.2 discusses the modification to the genetic operators laid out in references 6 and 7.

3.1 Geometry Encoding

In contrast to many previous geometry encoding schemes, the method used here is based on a genetic programming (*12*) implementation of constructive solid geometry. In addition, the primitives used to generate geometries are arbitrary convex polygons as defined by the convex hull of a list of points (*6*). Some methods in use today have adopted a genetic programming approach, but use a database of primitives (*13, 14*). In subsection 3.1.1, the convex polygon primitives are discussed, and in subsection 3.1.2, the combination scheme that accounts for inhomogeneous materials is reviewed from reference 8.

3.1.1 Convex Polygon Primitives

A convex polygon can be defined many ways, most simply as a polygon in which any line segment with both end points interior to the polygon never cross a boundary. A useful way to specify a convex polygon (for our purposes, especially) is as the convex hull of a set of points.

The convex hull of a set of points can be defined as the intersection of all half-planes that contain those points, or equivalently, as

$$H_{\text{convex}}(X) = \left\{ \sum_{i=1}^{k} \alpha_i \mathbf{x}_i \;\middle|\; \mathbf{x}_i \in X,\, \alpha_i \geq 0,\, \sum_{i=1}^{k} \alpha_i = 1,\, i = 1, 2, \ldots \right\}, \tag{17}$$

where X is some set of points and k varies from 1 to the total number of points in X. Several algorithms exist to compute the convex hull of a list of points, typically these algorithms return the subset of vertices that define the convex hull. There are different algorithms of varying computational complexity, fortunately $O(n \log n)$ and $O(n \log h)$ (where h is the number of vertices on the convex hull) algorithms exist (*15*).

Convex polygons are ideal in some ways for optimization. First, they are easy to define using an arbitrary list of points as described previously. Second, they have few constraints, i.e., any list of three or more, non-collinear points (in two dimensions) defines a convex polygon via its convex hull. Third, in this implementation, the number of points in a list is not restricted, so that they are flexible in "smoothness." Convex polygons can range from simple, triangular shapes or arbitrarily "smooth" approximations of truly smooth shapes. (Of course, a linear interpolation scheme is used throughout, but because the number of points is not restricted, they can approach smooth shapes such as circles and other curves.) Finally, they are easy to manipulate via genetic operators, especially crossover, as is discussed below. Using a database method can be difficult in terms of crossover, because it may not make sense to combine the defining parameters of a rectangle (width and height) with a circle (radius).

3.1.2 Inhomogeneous Combinations

While convex polygons are useful for optimization, on their own they are not sufficient to represent any arbitrary geometry or topology. Originally, Boolean combinations of convex polygons were used to generate more complex structures (*6*); unfortunately, Boolean combinations can only represent homogeneous structures. In this report, we wish to design multi-phase systems, so we must generalize the method in reference 6 to inhomogeneous systems, as in reference 8.

There are several ways to generalize the Boolean approach to inhomogeneous structures; here, a priority-based approach is used. Simply stated, each terminal node in a tree is assigned a priority value (some real number, initially randomly generated, between 1 and 10, for example) and material properties. Function nodes in this scheme are not of any specific type, the priority values dictate the specific operation; however, they are always binary. To decode, a tree is

traversed until a fully terminated subtree is reached. The priority values of the two terminal nodes (at this point they must be convex polygons) are rounded to the nearest integer and compared, and the polygon with the higher priority value is overlayed on top of the lower polygon. If the two priority values are equal, then a union operation is performed and the material values of the node on the left of the tree are used. The priority values of the resulting polygons are preserved for future operations.

Figure 3 illustrates the decoding process. Figure 3a shows each point list plotted in the plane with their convex hulls as the dashed lines. The points in each list are represented as symbols corresponding to the terminal nodes in figure 3b. Figure 3b shows the tree structure defining the priority combinations. In this example, material values are represented with shading. Figure 3c shows the result of the leftmost subtree. In this case, the priority values are equal when rounded, so the result is the union of the two convex polygons. Finally, figure 3d shows the overlap of the rightmost terminal node and the result of the previous step. The final result is an example of an inhomogeneous structure, which cannot be encoded using only Boolean operations. Now that the encoding/decoding scheme is defined, a set of genetic operators can be defined to guide a population of such tree chromosomes to a solution.

3.2 Genetic Operators

Central to any flavor of genetic algorithm or genetic programming method, the genetic operators, selection, crossover, and mutation can be adapted to the specific encoding/decoding scheme. As selection does not involve chromosome structure directly, standard techniques are used (*16*). Crossover involves hybridizing two encoded chromosomes by swapping subtrees and point lists (*6*) and is discussed in detail in subsection 3.2.1. Mutation, presented in subsection 3.2.2, increases genetic diversity and several different types are implemented. Many have been discussed in references 6–8, though some have been altered here.

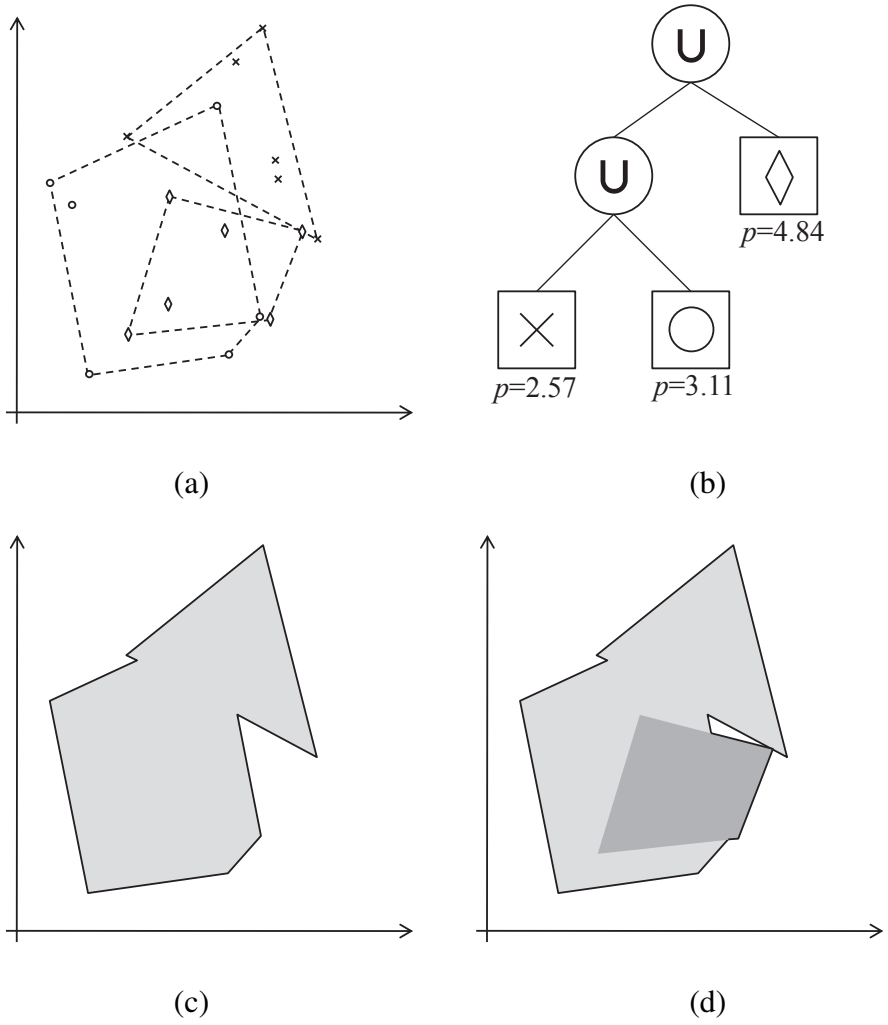

Figure 3. Example of the decoding process: (a) Point lists with their convex hulls, (b) tree defining priority combinations, (c) first subtree decoded, and (d) final result .

3.2.1 Crossover

Crossover is the hybridization of two parent chromosomes resulting in two children, which, at times, contain good traits from each parent. Typically, crossover involves splicing parts of each chromosome between parents, and possibly hybridizing some nodes. For genetic programming, subtrees are exchanged and the root nodes of those subtrees may be hybridized in some way. Crossover for genetic programming can be implemented in a straightforward way; however, "tree bloat"—the exponential growth of chromosomes—can be a major cause of population stagnation (*17*). A solution to tree bloat is to design a crossover scheme that has less variation than the standard implementation. In other words, the children should resemble their parents closely. In reference 8, this was accomplished two ways. First, a geometric (phenotypic) similarity-based crossover probability was used, and also only terminal nodes were used in crossover. Here, the first method is retained, while the second is abandoned. Only involving terminal nodes in crossover will certainly eliminate tree bloat via crossover, though it may be too restrictive for more complicated topologies. Instead, the population is initialized only with single-node trees (i.e., a single convex polygon), and mutation is primarily responsible for altering the tree structure.

Selecting a chromosome for crossover now involves two steps. First, for a given chromosome, a potential set of mates is chosen at random. The number of potential mates is set at

$$N_{\mathrm{m}} = \lceil N_{\mathrm{p}}/20 \rceil \,, \tag{18}$$

where N_{p} is the population size. For each potential mate, a set of random subtrees are chosen and compared. The number of subtrees to compare for crossover is an input parameter N_{ca}, typically set to 10. (Of course, in some cases there may be fewer than N_{ca} unique combinations of subtrees, so only the smaller of N_{ca} and the total unique combinations are compared.) Regardless, each chosen subtree is decoded and the resulting geometries are compared for similarity using

$$p_{\mathrm{o}} = \left(\frac{2a_{\mathrm{i}}}{a_1 + a_2} \right)^s , \tag{19}$$

where a_1 and a_2 are the areas of the decoded subtrees, a_{i} is the area of their intersection, and s is some given biasing exponent, typically chosen as 2.5. The subtree combination with the highest p_{o} is saved and used as the similarity for that pair of mates. Once the similarity of all mates is computed, a crossover mate is chosen using roulette wheel selection (*16*), and a final crossover probability is computed

$$p_{\mathrm{m}} = p_{\mathrm{c}} p_{\mathrm{o}}, \tag{20}$$

where p_c is a given crossover probability, typically between 0.9 and 1. Finally, a weighted coin is flipped with probability p_m, and if successful, the subtree pairs chosen previously are used for crossover. If two terminal nodes are chosen, their point lists are hybridized.

In addition to tree bloat, the point lists in each terminal node are not subject to a maximum size, so they can also grow exponentially. To prevent this, point lists are hybridized using two-point linear crossover, and the number of points to swap—while still random—is kept constant between each point list. Swapping an equal number of points between two point lists ensures that neither point list changes size during crossover. Also, the points located at each crossover site are hybridized by taking a random, weighted sum of the components of each point, as is standard in many real-coded genetic algorithm implementations. Priority values are hybridized in the same way, and currently, material values are not changed as the materials are chosen from a database.

3.2.2 Mutation

The main purpose of mutation is to insert new genetic information into a population at random, hopefully reducing the chance of premature convergence. Given the complexity of the chromosome used here and its geometric meaning, there are many different types of mutations possible. These have been discussed extensively in references 6–8, so only new mutations are covered here. One major difference in general is that, wherever possible, Gaussian random variables are now used as perturbations rather than uniform random variables in some specified range.

The first new mutation is a point addition mutation, though rather than randomly adding a point anywhere in the valid region, the point is added along a line segment of the convex hull. Given two points, a point can be generated lying at a random point between them

$$\mathbf{p}^* = t\mathbf{h}_1 + (1-t)\mathbf{h}_2, 0 \leq t \leq 1, \tag{21}$$

where t is a uniform random variable and $\mathbf{h}_1$ and $\mathbf{h}_1$ are neighboring points on a convex hull. Adding $\mathbf{p}^*$ to the point list will not affect the decoded shape, but it may be useful in future generations. In a similar vein, the point addition mutation has been altered and no longer simply adds a random point. Point addition proceeds by duplicating a randomly chosen point and translating it by a small amount. Previously, it was observed that the point addition mutation was destructive, especially in problems with a large region size. Duplicating and translating an existing point is less severe.

Terminal node splitting, an important mutation operation, alters the topology of the tree and

geometry. Previously, terminal nodes were split in half (*6*) and at the center (*8*). In this implementation, a third possible split is used, which simply separates the terminal node's point list into two new terminal nodes. The split site is chosen at random, and the priority and material values of the original terminal node are preserved. This operation can be useful for generating new geometries, as the previous were designed to generate an identical geometry when decoded.

Finally, priority values and material values can be mutated. Priority values are mutated by adding a Gaussian random variable with a variance of 2 and a zero mean. The decoded chromosome may not change with a change in priority (depending on the priority values of the other nodes) so this can be used with a high rate. Materials are mutated simply by randomly choosing a new material from the database.

In its entirety, the genetic programming method presented here is similar to references 6–8, but with a few important changes. These changes lead to a more efficient and effective method as is demonstrated in section 4.

---

4. Results

The following are a few examples of the method described previously using different materials. The following examples were run on a parallel computing system, each using 20, four-core computers giving a total of 80 nodes. Input parameters for each example are given in table 1.

4.1 Example 1

In the first example, two different materials were available to the algorithm: epoxy and lead, with material properties taken from reference 2 and epoxy fixed as the matrix. A square unit cell was used, with side lengths of 1 m. The best result found, shown in figure 4 with light gray indicating epoxy and dark gray indicating lead, has a relative bandgap size of 0.68 and required 260 generations. Figure 5 gives the band diagram for this example, and the center frequency of the bandgap is about 850 Hz.

Table 1. Genetic programming input parameters.

Parameter		**Value**
Population	Size	300
	Tournament Size	4
	Crossover Rate	85%
	Elitism	on
Mutation	Delete nodes	0.5%
	Delete one point	2%
	Add duplicate point	0.5%
	Split hull segment	0.5%
	Geometry transform	1%
	Priority	1%
	Material	1%
	Prune	5%
	Hull	1%
	Point translation	1%
	Split in half	1%
	Split a hole	1%
	Split list	1%
	Combine	6%

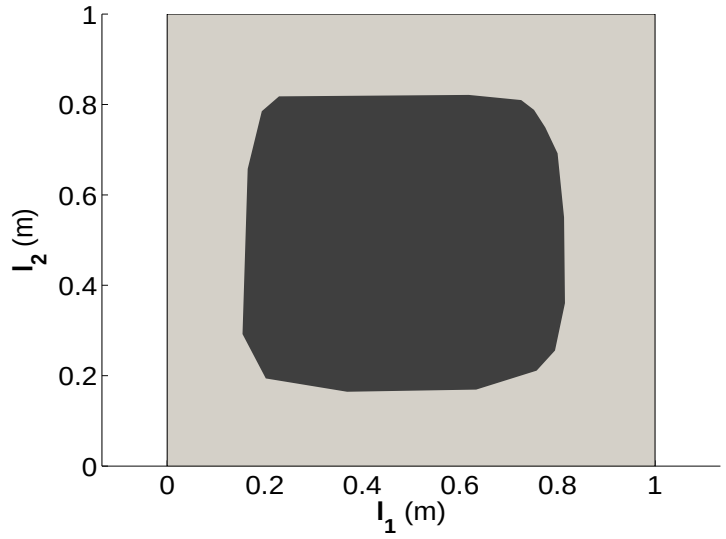

Figure 4. Results of example 1; light gray indicates epoxy and dark gray indicates lead.

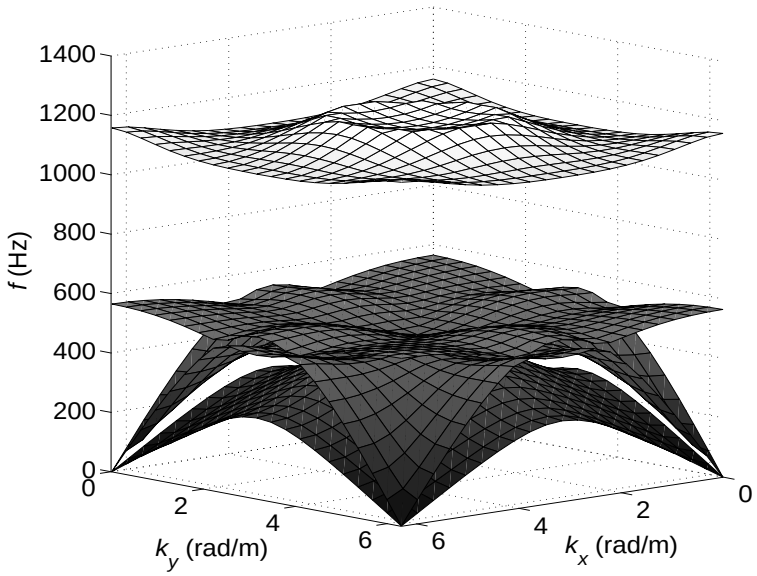

Figure 5. Band diagram for example 1.

4.2 Example 2

Next, a third material was added, silicone rubber, again with material properties taken from reference 2; however, the algorithm was allowed to choose the background material. These materials were chosen in reference 2 to construct a low frequency, local resonance based bandgap, with a design of a silicone rubber-coated lead sphere embedded in an epoxy matrix laid out in a cubic arrangement. Again, a square unit cell was used with side lengths of 1 m. A bandgap size of 1.56 was achieved after 565 generations. The best result found is shown in figure 6. In contrast to the design in reference 2, the genetic programming-generated design does not coat the entire mass of lead in silicone rubber, though it still shares common features such as an epoxy background with lead embedded in silicone rubber. The band diagram for this design is shown in figure 7, which shows the center frequency of the band is around 550 Hz.

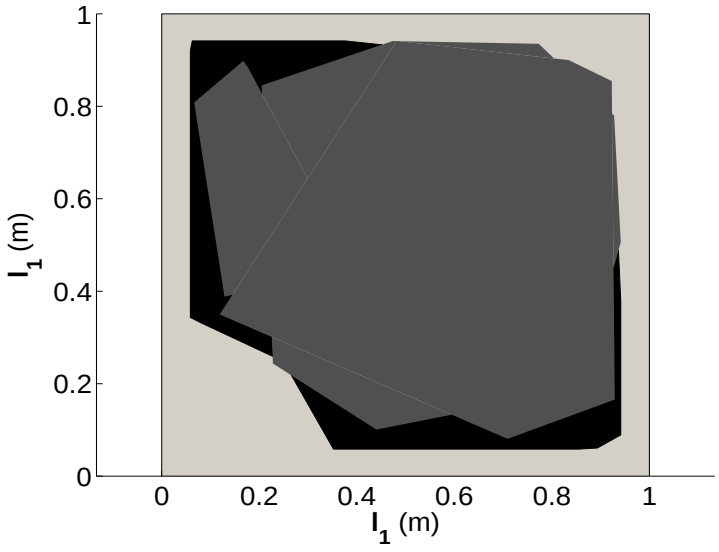

Figure 6. Results of example 2; light gray indicates epoxy, dark gray indicates lead, and black represents silicone rubber.

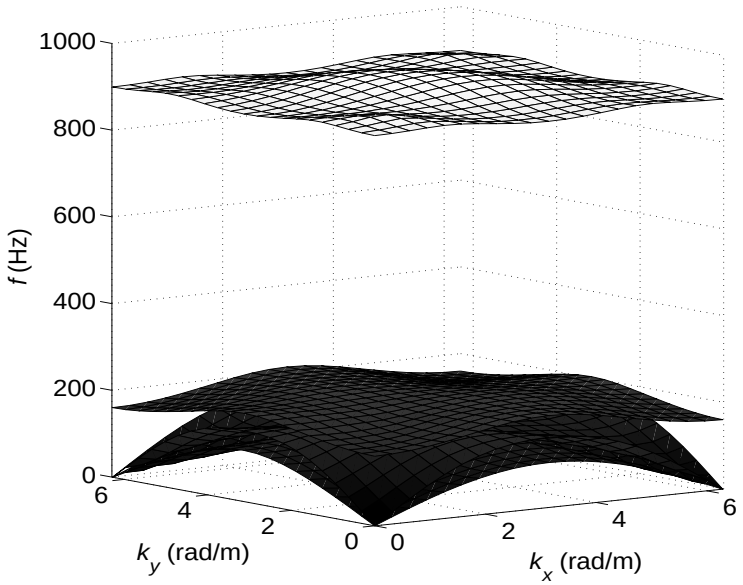

Figure 7. Band diagram for example 2.

5. Conclusions

A genetic programming method was applied to design the inclusion of phononic bandgap materials to maximize the bandgap size. The geometry was encoded using a tree structure that combines convex polygons using a priority-based overlapping scheme. The algorithm could choose from a database of materials for both the inclusion and the matrix. Results were given for three materials: lead, silicone rubber, and epoxy. One example restricted the method to use only lead and epoxy with epoxy fixed as the matrix material. In another example, all three materials were allowed with no fixed matrix. Bandgaps were achieved for each material system.

In the future, the method could be extended to a multi-objective problem, where bandgap size could be simultaneously optimized with another goal such as center frequency or inclusion size. Additional materials could be used for specific applications.

6. References

1. Wang, Y.; Li, F.; Wang, Y.; Kishimoto, K.; Huang, W. Tuning of band gaps for a two-dimensional piezoelectric phononic crystal with a rectangular lattice. *Acta Mech. Sin.* **2009**, *25* (1), 65–71.

2. Liu, Z.; Zhang, X.; Mao, Y.; Zhu, Y.; Yang, Z.; Chan, C.; Sheng, P. Locally resonant sonic materials. *Science* **2000**, *289* (5485), 1734–1736.

3. Kushwaha, M. S.; Halevi, P. Band-Gap Engineering in Periodic Elastic Composites. *Appl Phys. Lett.* **1994**, *64* (9), 1085–1087.

4. Sigmund, O.; Jensen, J. Systematic design of phononic band-gap materials and structures by topology optimization. *Proc. R. Soc. A-Math. Phys. Eng. Sci.* **2003**, *361* (1806), 1001–1019.

5. Gazonas, G. A.; Weile, D. S.; Wildman, R. A.; Mohan, A. Genetic algorithm optimization of phononic bandgap structures. *Int. J. Solids Struct.* **2006**, *43* (18–19), 5851–5866.

6. Wildman, R. A.; Weile, D. S. Geometry reconstruction of conducting cylinders using genetic programming. *IEEE T. Antennas Propag.* **2007**, *55* (3), 629–636.

7. Wildman, R. A.; Weile, D. S. Greedy search and a hybrid local optimization/genetic algorithm for tree-based inverse scattering. *Microw. Opt. Technol. Lett.* **2008**, *50* (3), 822–825.

8. Wildman, R. A.; Weile, D. S. Inverse Scattering of Dielectric Cylindrical Targets Using Genetic Programming. *Electromagnetics* **2010**, *30* (1–2, Sp. Iss. SI), 222–236.

9. Brillouin, L. *Wave Propagation in Periodic Structures, Electric Filters and Periodic Lattices*; McGraw-Hill: New York, NY, 1946.

10. Persson, P.; Strang, G. A simple mesh generator in MATLAB. *SIAM Rev.* **2004**, *46* (2), 329–345.

11. Dennis, J. E.; Schnabel, R. B. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*; Society for Industrial and Applied Mathematics: Philadelphia, PA, 1996.

12. Koza, J. R. *Genetic programming III: Darwinian invention and problem solving*; Morgan Kauffman, Inc: San Francisco, CA, 1999.

13. Yamagiwa, M.; Uehara, M.; Murakami, M.; Yoneyama, M. A reconstruction method for ultrasonic deterioration image by the combination of constructive solid geometry and strongly typed genetic programming. *Comput. Syst. Sci. Eng.* **2010**, *25* (2, Sp. Iss. SI), 161–170.

14. Hamza, K.; Saitou, K. Optimization of constructive solid geometry via a tree-based multi-objective genetic algorithm. In *Genetic And Evolutionary Computation Gecco 2004 , Pt 2, Proceedings*; Vol. 3103; Deb, K. and Poli, R. and Banzhaf, W. and Beyer, H. G. and Burke, E. and Darwen, P. and Dasgupta, D. and Floreano, D. and Foster, O. and Harman, M. and Holland, O. and Lanzi, P. L. and Spector, L. and Tettamanzi, A. and Thierens, D. and Tyrrell, A., Ed.; Springer-Verlag Berlin: Berlin, Germany, 2004; pp 981–992.

15. de Berg, M.; van Kreveld, M.; Overmars, M.; Schwarzkopf, O. *Computational Geometry: Algorithms and Applications*; 2nd ed.; Springer: New York, NY, 2000.

16. Goldberg, D. E. *Genetic Algorithms in Search, Optimization, and Machine Learning*; Addison-Wesley: Reading, MA, 1987.

17. Banzhaf, W.; P., N.; Keller, R. E.; D., F. F. *Genetic programming: An introduction*; Morgan Kauffman, Inc: San Francisco, CA, 1999.

NO. OF COPIES	ORGANIZATION
1 (PDF ONLY)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA 8725 JOHN J KINGMAN RD STE 0944 FORT BELVOIR VA 22060-6218
1	DIRECTOR US ARMY RESEARCH LAB IMNE ALC HRR 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB RDRL CIO LL 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB RDRL CIO MT 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB RDRL D 2800 POWDER MILL RD ADELPHI MD 20783-1197

NO. OF COPIES	ORGANIZATION
2	NSF S MCKNIGHT G PAULINO 4201 WILSON BLVD, STE 545 ARLINGTON, VA, 22230-0002
2	DARPA W COBLENZ J GOLDWASSER 3701 N FAIRFAX DR ARLINGTON VA 22203-1714
1	DIRECTOR US ARMY ARDEC AMSRD AAR AEE W E BAKER BLDG 3022 PICATINNY ARSENAL NJ 07806-5000
2	US ARMY TARDEC AMSTRA TR R MS 263 K BISHNOI D TEMPLETON MS 263 WARREN MI 48397-5000
1	COMMANDER US ARMY RSRCH OFC RDRL ROI M J LAVERY PO BOX 12211 RESEARCH TRIANGLE PARK NC 27709-2211
1	COMMANDER US ARMY RSRCH OFC RDRL ROE M D STEPP PO BOX 12211 RESEARCH TRIANGLE PARK NC 27709-2211
6	NAVAL RESEARCH LAB E R FRANCHI CODE 7100 M H ORR CODE 7120 J A BUCARO CODE 7130 G J ORRIS 7140 J S PERKINS CODE 7140 S A CHIN BING CODE 7180 4555 OVERLOOK AVE SW WASHINGTON DC 20375

NO. OF COPIES	ORGANIZATION
1	DTRA M GILTRUD 8725 JOHN J KINGMAN RD FORT BELVOIR VA 22060
1	ERDC US ARMY CORPS OF ENGINEERS USACEGSL P PAPADOS 7701 TELEGRAPH RD ALEXANDRIA VA 22315
1	AFOSR/NL 875 NORTH RANDOLPH ST SUITE 325, RM 3112 F FAHROO ARLINGTON VA 22203
1	CLEMSON UNIV DEPT MECH ENGINEERS M GRUJICIC 241 ENGRG INNOVATION BLDG CLEMSON SC 29634-0921
1	UNIV OF CALIFORNIA CTR OF EXCELLENCE FOR ADV MATLS S NEMAT NASSER SAN DIEGO CA 92093-0416
3	DIRECTOR LANL P MAUDLIN A ZUREK F ADDESSIO PO BOX 1663 LOS ALAMOS NM 87545
7	DIRECTOR SANDIA NATL LABS J BISHOP MS 0346 E S HERTEL JR MS 0382 W REINHART MS 1181 T VOGLER MS 1181 L CHHABILDAS MS 1811 M FURNISH MS 1168 M KIPP MS 0378 PO BOX 5800 ALBUQUERQUE NM 87185-0307

NO. OF COPIES	ORGANIZATION
1	DIRECTOR LLNL M J MURPHY PO BOX 808 LIVERMORE CA 94550
3	CALTECH M ORTIZ MS 105 50 G RAVICHANDRAN T J AHRENS MS 252 21 1201 E CALIFORNIA BLVD PASADENA CA 91125
5	SOUTHWEST RSRCH INST C ANDERSON K DANNEMANN T HOLMQUIST G JOHNSON J WALKER PO DRAWER 28510 SAN ANTONIO TX 78284
1	TEXAS A&M UNIV DEPT OF MATHEMATICS J WALTON COLLEGE STATION TX 77843
1	UNIVERSITY OF MISSISSIPPI DEPT OF MECH ENGRG A M RAJENDRAN 201-B CARRIER HALL UNIVERSITY, MS 38677
2	SRI INTERNATIONAL D CURRAN D SHOCKEY 333 RAVENSWOOD AVE MENLO PARK CA 94025
1	VIRGINIA POLYTECHNIC INST COLLEGE OF ENGRG R BATRA BLACKSBURG VA 24061-0219
7	UNIV OF NEBRASKA DEPT OF ENGRG MECH F BOBARU Y DZENIS G GOGOS M NEGAHBAN R FENG J TURNER Z ZHANG LINCOLN NE 68588

NO. OF COPIES	ORGANIZATION
1	JOHNS HOPKINS UNIV DEPT OF MECH ENGRG K T RAMESH LATROBE 122 BALTIMORE MD 21218
1	WORCESTER POLYTECHNIC INST MATHEMATICAL SCI K LURIE WORCESTER MA 01609
4	UNIV OF UTAH DEPT OF MATH A CHERKAEV E CHERKAEV E S FOLIAS R BRANNON SALT LAKE CITY UT 84112
1	PENN STATE UNIV DEPT OF ENGRG SCI & MECH F COSTANZO UNIVERSITY PARK PA 168023
4	UNIV OF DELAWARE DEPT OF MECH ENGRG T BUCHANAN T W CHOU A KARLSSON M SANTARE 126 SPENCER LAB NEWARK DE 19716
1	UNIV OF DELAWARE CTR FOR COMPST MATRLS J GILLESPIE NEWARK DE 19716
1	COMPUTATIONAL MECH CONSULTANTS J A ZUKAS PO BOX 11314 BALTIMORE MD 21239-0314
1	LOUISIANA STATE UNIV R LIPTON 304 LOCKETT HALL BATON ROUGE LA 70803-4918
1	INST OF ADVANCED TECH UNIV OF TX AUSTIN S BLESS 3925 W BRAKER LN STE 400 AUSTIN TX 78759-5316

NO. OF COPIES ORGANIZATION

1 APPLIED RSCH ASSOCIATES
D E GRADY
4300 SAN MATEO BLVD NE
STE A220
ALBUQUERQUE NM 87110

1 INTERNATIONAL RSRCH
ASSOC INC
D L ORPHAL
4450 BLACK AVE
PLEASANTON CA 94566

2 WASHINGTON ST UNIV
INST OF SHOCK PHYSICS
Y M GUPTA
J ASAY
PULLMAN WA 99164-2814

1 NORTHWESTERN UNIV
DEPT OF CIVIL & ENVIRON ENGRG
Z BAZANT
2145 SHERIDAN RD A135
EVANSTON IL 60208-3109

1 UNIV OF DAYTON
RSRCH INST
N S BRAR
300 COLLEGE PARK
MS SPC 1911
DAYTON OH 45469

2 TEXAS A&M UNIV
DEPT OF GEOPHYSICS MS 3115
F CHESTER
T GANGI
COLLEGE STATION TX 778431

1 UNIV OF SAN DIEGO
DEPT OF MATH & CMPTR SCI
A VELO
5998 ALCALA PARK
SAN DIEGO CA 92110

1 NATIONAL INST OF
STANDARDS & TECHLGY
BLDG & FIRE RSRCH LAB
J MAIN
100 BUREAU DR MS 8611
GAITHERSBURG MD 20899-8611

NO. OF COPIES ORGANIZATION

1 MIT
DEPT ARNTCS ASTRNTCS
R RADOVITZKY
77 MASSACHUSETTS AVE
CAMBRIDGE MA 02139

1 UNIV OF DELAWARE
DEPT ELECTRICAL & CMPTR ENGRG
D WEILE
NEWARK DE 19716

1 T W WRIGHT
4906 WILMSLOW RD
BALTIMORE MD 21210

1 UNIV OF TEXAS-PAN AMERICAN
COLLEGE OF ENGRG
& COMPUTER SCI
D H ALLEN
1201 WEST UNIVERSITY DR
EDINBURG, TX 78539-2999

3 RDRL D
C CHABALOWSKI
J CHANG
R SKAGGS
BLDG 205
2800 POWDER MILL RD
ADELPHI MD 20783-1197

ABERDEEN PROVING GROUND

79 DIR USARL
RDRL WM
B FORCH
S KARNA
J MCCAULEY
P PLOSTINS
M ZOLTOSKI
RDRL WML
D LYONS
J NEWILL
RDRL WML B
I BATYREV
S IZVYEKOV
B RICE
N WEINGARTEN
RDRL WML D
P CONROY
M NUSCA
RDRL WML G
M BERMAN
W DRYSDALE

NO. OF COPIES	ORGANIZATION
	RDRL WML H
	D SCHEFFLER
	S SCHRAML
	B SCHUSTER
	RDRL WMM
	J BEATTY
	R DOWDING
	RDRL WMM A
	M MAHER
	J TZENG
	E WETZEL
	RDRL WMM B
	T BOGETTI
	B CHEESEMAN
	C FOUNTZOULAS
	G GAZONAS
	D HOPKINS
	P MOY
	B POWERS
	C RANDOW
	T SANO
	M VANLANDINGHAM
	R WILDMAN
	C F YEN
	RDRL WMM C
	J LA SCALA
	RDRL WMM D
	E CHIN
	K CHO
	RDRL WMM E
	J ADAMS
	M COLE
	T JESSEN
	J LASALVIA
	P PATEL
	J SANDS
	RDRL WMM F
	L KECSKES
	H MAUPIN
	RDRL WML G
	J ANDZELM
	A RAWLETT
	RDRL WMP
	P BAKER
	S SCHOENFELD
	RDRL WMP B
	R BECKER
	S BILYK
	D CASEM
	J CLAYTON
	M GREENFIELD
	C HOPPEL
	R KRAFT
	B LEAVY
	M RAFTENBERG
	S SATAPATHY
	M SCHEIDLER
	T WEERASOORIYA
	RDRL WMP C
	T BJERKE
	S SEGLETES
	RDRL WMP D
	R DONEY
	D KLEPONIS
	J RUNYEON
	B SCOTT
	H MEYER
	RDRL WMP E
	M BURKINS
	B LOVE
	RDRL WMP F
	M CHOWDHURY
	A FRYDMAN
	N GNIAZDOWSKI
	R GUPTA
	RDRL WMP G
	N ELDREDGE
	D KOOKER
	S KUKUCK
	G R PEHRSON

INTENTIONALLY LEFT BLANK.